

In the Specification:

Please amend the Abstract as follows:

ABSTRACT

A system and method for providing automatic and coordinated sharing of conversational resources, e.g., functions and arguments, between network-connected servers and devices and their corresponding applications. In one aspect, a system for providing automatic and coordinated sharing of conversational resources ~~inlcudes eompries:~~ a network ~~having eompriing~~ a first (100), and second (106) network device, ; the first (100) and second (106) network device each comprising a set of conversational resources (102, 107), a dialog manager (103, 108), for managing a conversation and executing calls requesting a conversational service, and a communication stack (111, 115), for communicating messages over the network using conversational protocols, wherein the conversational protocols establish coordinated network communication between the dialog managers of the first and second network device to automatically share the set of conversational resources of the first and second network device, when necessary, to perform their respective requested conversational service.

Please amend the paragraph on p. 7, lines 3-16 as follows:

In particular, to provide conversational coordination between the networked devices to share their conversational functions, resources and arguments, each of the networked devices communicate messages using conversational protocols (or methods) to exchange information regarding their conversational capabilities and requirements. For instance, as shown in Fig. 1, the client device 100 comprises a communication stack 111 for transmitting and receiving messages using conversational protocols 112, conversational discovery, registration and negotiation protocols 113 and speech transmission protocols 114 (or conversational coding protocols). Likewise, the server 106 comprises a server communication stack 115 comprising conversational protocols 116, conversational discovery, registration and negotiation protocols 117 and speech transmission protocols 118. These protocols (methods) are discussed in detail with respect to a CVM (conversational virtual machine) in the patent application IBM Docket No. YO999-111P, filed concurrently herewith, entitled "Conversational Computing Via Conversational Virtual Machine," (i.e., International Appl. No. PCT/US99/22927, filed on October 1, 1999 and corresponding U.S. Patent Application Serial No. 09/806,565) which is commonly assigned and incorporated herein by reference.

Please amend the paragraph on p. 22, line 28 ~ p. 24, line 3, as follows:

The system of Fig. 1 may also be implemented with the conversational browser system described in IBM Docket No. YO998-392P, filed concurrently herewith, entitled "Conversational Browser and Conversational Systems," (i.e., International Appl. No. PCT/US99/23008, filed on October 1, 1999 and corresponding U.S. Patent Application Serial No. 09/806,544), which is commonly assigned and incorporated herein by reference, wherein a CML (conversational markup language) page, which is similar in concept to HTML (hypertext markup language) page for visual display, transmitted from a content provider (server) (and processed by the conversational browser) is used to describe a conversational UI to be presented to the user. In this example, the conversational browser can be the local application 104 of the client device 100 and/or the server application 109 in the remote (IVR) server 106. It is possible for the content provider or application developer (or the proxy/transcoder) to decide that a given item that the user should provide (for example a NLU or FSG input of a form or a free form to fill via dictation) must be recognized on the server 106 rather than furnishing all the data to the client device 100 for local recognition (because the task is too complex for the local resources or because too much information would have to be sent through the network). This is done, for example, by providing a URL (uniform resource locator) and tags in a CML file to indicate a sever in which the processing will occur, or by loading in the CML page an applet, an Active X component or a plug-in (or whatever variation of it) which capture the audio, performs possibly some conversational functions and ship it to other devices for other functions (this is typically a decision made by the author of the page). This decision can be automatically performed by a transcoder and a registration mechanism as described in IBM Docket No. YO998-392P, whereby the browser explicitly describes its capability to the server where it fetches the CML page. When the transcoder is used to additionally take into account the capabilities of the browser and adapt the content to such capabilities (this capability is what is referred to a conversational proxy), the transcoder can add the server URL(s) to redirect the server now on the basis of the browser's capabilities. In such cases, the speech collected by the client device 100 can be sent as a waveform (compressed or not) or as stream of features to either the remote server 106 or the networked server 110 where the recognition occurs (or NLU/NLG). The recognition result can

then be sent back to the client device 100 or to the CML provider server (remote server 106) to decide the next course of action or further processing. Again as mentioned above, this can be decided by the application that can directly incorporate the URL of the resource/engine/server or local device to be used for the recognition of a given input, menu form or dialog. In addition, the present invention is useful in circumstances where a CML page must playback/synthesize sounds or text too complex for the local conversational engines 102 of the local device 102. The portions that are too complex may be obtained as stream of features or compressed waveforms from either a specific server (which may or may not be the server that provided the CML page). Furthermore, with regard to multi-lingual systems, if a CML page involves a different language, the local client device 100 without the appropriate capability can request a remote server to perform the conversational functions in that language.

Please amend the paragraph on p. 1, lines 4-5, as follows:

Cross-Reference to Related Applications

This application is a U.S. National Phase Application filed under 35 U.S.C. 371 based on International Application No. PCT/US99/22925, filed on October 1, 1999, which is based on provisional applications U.S. Serial Number 60/102,957, filed on October 2, 1998, and U.S. Serial No. 60/117,595 filed on January 27, 1999.